A

Project Report

on

"Comprehensive Analysis of AI Analytics Platforms, Competitive Evaluation and Automation through Python based Selenium Framework - **Enhancing Business Intelligence through Automated Python based Selenium Framework"**

at

WingsBI Technology and Solutions Private Limited

Conducted By

Mr. Dipak Patle

For partial fulfilment of Course No. 307 of MBA Syllabus

(2023-2025)

of

Savitribai Phule Pune University

During May to July 2024

Under the Guidance of

Dr. Suwarna Shidore (Internal Guide)

IndSearch Institute of Management Studies & Research

Pune 411021

December 2024

DECLARATION

I hereby declare that the project report entitled **"Comprehensive Analysis of AI Analytics Platforms, Competitive Evaluation and Automation through Python based Selenium Framework "** with special reference to **“WingsBI Technology and Solutions Private Limited”** has been written and submitted by me. The project consists of three major modules. The module titled **Enhancing Business Intelligence through Automated Python based Selenium Framework** is designed and developed by me. The period of project was three months starting from May, 2024. This project is submitted to Savitribai Phule Pune University as a partial fulfilment of requirements for the award of degree of Master of Business Administration under the guidance of Dr. Suwarna Shidore.

I further declare that this project report is neither produced from any other document nor it is submitted to any other company or University.

All the sources have been duly acknowledged in this report.

Date: Mr. Dipak Narendra Patle

Place: Pune PRN: 23255109

ACKNOWLEDGMENT

The internship opportunity I had with **"WingsBI Technology & Solutions Private Limited, Pune"** was a great chance for learning and professional development. Therefore, I consider myself a very lucky individual to have been provided with an opportunity to be a part of it. I am also grateful for having the chance to meet so many wonderful people and professionals who guided us throughout this internship period.

It's a great pleasure to express my heartfelt thanks to our Director, Dr. Aparna Tembulkar, and Dean, Dr. Suwarna Shidore, for allowing us to carry out this training at WingsBI Technology & Solutions Private Limited, Pune. I would also like to extend my gratitude to Mr. Ajit Pawar, Mr. Ravi Kakade, and Mr. Shreyas Pande for their guidance and support during this internship.

I perceive this opportunity as a significant milestone in my career development. I will strive to use the skills and knowledge I have gained in the best possible way and continue working towards further improvement.

Dipak Narendra Patle
Student, MBA II
Date:
Place: Pune

Table of Contents

Executive Summary

The project titled "Enhancing Business Intelligence through Automated Python-based Selenium Framework" is designed to transform the way businesses interact with data and generate insights. By harnessing the power of automation, the project aims to streamline the entire data processing and reporting lifecycle, improving both the accuracy and efficiency of business intelligence (BI) tasks. This is achieved by using a robust combination of Python and Selenium to automate data collection, response analysis, report generation, and comparison of results from multiple AI analytics platforms.

At its core, the project seeks to automate several key processes that are traditionally time-consuming and prone to human error. One of the main objectives is to automate the processing of questions and the generation of outputs based on interactions with AI-powered analytics tools. This eliminates the need for manual input and significantly reduces the time spent on repetitive tasks. The automation framework ensures that questions are consistently processed, with responses automatically captured and saved in a structured and organized manner. These responses are stored as PNG images, providing a visual representation of the data that can be easily referenced and accessed when needed.

Beyond simple data collection, the project introduces a comparison mechanism that allows for the evaluation of responses from different AI analytics products. By automating this comparison, the system highlights discrepancies and variations in the data, offering businesses valuable insights into the performance and reliability of the different AI tools. This comparison helps identify patterns, outliers, or inconsistencies, enabling organizations to make more informed decisions based on a clearer understanding of the results produced by various analytics solutions.

Another key aspect of the project is the automated generation of PowerPoint (PPT) reports. Instead of manually compiling data into presentation slides, the system automatically creates comprehensive, visually appealing reports that include key insights, comparisons, and performance metrics. This greatly reduces the time needed to

produce professional-quality presentations, making it easier for decision-makers to review and act on the information without delays.

The project also incorporates score calculation and reporting. The automation framework not only captures and compares data but also evaluates performance by calculating scores based on predefined criteria. These scores are automatically incorporated into the final report, providing a clear summary of the results and enabling businesses to gauge performance against set benchmarks or KPIs (Key Performance Indicators).

One of the most important benefits of the project is the enhanced accuracy and consistency of the data. By automating the entire process, the system minimizes the risk of human error and ensures that all steps are executed according to a standardized procedure. This results in more reliable data, which in turn improves the overall quality of business insights. Furthermore, the automation framework is designed to be flexible and scalable, allowing it to evolve with changing business requirements. As new data sources or analytics tools are introduced, the system can be easily adapted to accommodate these changes without requiring significant rework or redevelopment.

In addition to improving data quality and reducing errors, the project also focuses on improving efficiency. By automating repetitive tasks such as data collection, response comparison, and report generation, the system helps organizations significantly reduce the time spent on manual processes. This not only accelerates the time-to-insight but also frees up valuable resources to focus on more strategic activities. The result is a more agile, responsive business environment where decisions can be made faster and with greater confidence.

In summary, the part of the project "Enhancing Business Intelligence through Automated Python-based Selenium Framework" provides a comprehensive solution to automate the key processes involved in business intelligence. By integrating Python and Selenium, the system enables businesses to capture, compare, and report data more efficiently, accurately, and consistently. With its ability to automate time-consuming tasks, reduce errors, and improve decision-making speed, the project offers a powerful tool to enhance the quality and timeliness of business insights, ultimately driving better business outcomes and fostering a more data-driven approach to decision-making.

Chapter I: Introduction

Introduction

In today's fast-paced, data-driven business environment, organizations are increasingly relying on Business Intelligence (BI) tools to make informed decisions, optimize operations, and gain competitive advantages. As businesses accumulate vast amounts of data, the challenge of turning that data into actionable insights becomes more complex. BI platforms such as Zoho Analytics and WingsBI offer powerful capabilities for data analysis and visualization, enabling organizations to track performance, identify trends, and make more accurate forecasts. However, while these platforms are robust, fully leveraging their capabilities requires overcoming significant challenges—chief among them being the time and effort required for manual data collection, reporting, and decision-making. In many cases, manually querying the platforms, collecting data, and preparing reports can be tedious, error-prone, and inefficient.

As organizations strive for operational efficiency and better decision-making, automation has emerged as a key enabler for enhancing the Business Intelligence (BI) process. Automation enables the handling of repetitive tasks—such as querying databases, extracting insights, generating reports, and comparing results—without the need for continuous human intervention. By automating these workflows, businesses can significantly reduce the time spent on data collection and analysis, improve the accuracy of their reports, and focus more resources on data interpretation and strategic decision-making. This project aims to demonstrate how automation, combined with Python and the Selenium framework, can significantly enhance the BI process by automating the querying, result capturing, and report generation tasks across two leading BI platforms: Zoho Analytics and WingsBI.

Project Objective

The core objective of this project was to develop an automated framework that simplifies and accelerates the process of extracting insights from BI platforms using "**Exploring the AI Analytics Market and Developing Databases Using PostgreSQL"** and **"AI-Powered Insights: Competitive Evaluation of Leading Analytics Platforms"** which is done by my colleague. Through the automation of key tasks, this project enhances the effectiveness of BI workflows, improves the consistency

of results, and generates detailed, actionable insights with minimal manual effort. The framework aims to automate several key activities: logging into the Zoho Analytics and WingsBI platforms, querying predefined questions from a CSV file, capturing the corresponding output as screenshots, and creating a comprehensive comparative report in the form of a PowerPoint presentation.

The first step in the automation process involves securely logging into the Zoho Analytics and WingsBI platforms using predefined user credentials. Once logged in, the framework proceeds to interact with the dashboards of both platforms, initiating the importation of the required datasets or databases provided by the organization's higher entities. These databases often contain essential business data that needs to be analyzed and visualized to uncover trends, patterns, and insights. After successfully importing the data, the script transitions to the "Ask Questions" page, where the framework queries predefined questions from a CSV file. These questions are designed to elicit specific insights from both platforms, which are essential for reporting and decision-making.

Automated Query Execution and Data Capture

For each question in the CSV file, the automated script interacts with both Zoho Analytics and WingsBI, querying the platforms to generate corresponding outputs. These queries range from basic statistical questions to more complex analytical tasks, such as filtering data, aggregating metrics, and comparing results across different datasets. Once the output is generated, the automation script captures a screenshot of the result. The screenshot serves as a visual representation of the query result, which is then saved with a unique file name corresponding to the question asked. This file is organized into separate folders for each BI tool—Zoho Analytics and WingsBI—to ensure clarity and ease of access.

The decision to capture screenshots rather than extracting raw data directly is intentional, as it allows for the visual comparison of outputs from both platforms. These images reflect the visualizations, charts, and reports generated by each platform, making it easier for decision-makers to understand and compare the results. Moreover, capturing screenshots provides a layer of documentation, ensuring that all data and results are preserved for future reference or review.

Presentation Generation and Image Comparison

Once all questions have been processed, the next major task in the automation workflow is to generate a PowerPoint presentation that consolidates the results. The PowerPoint slides are designed to feature both the questions from the CSV file and the corresponding screenshots from Zoho Analytics and WingsBI, side by side. This enables a direct visual comparison of the results produced by each platform. The purpose of this comparative analysis is to allow decision-makers to quickly assess which platform provides more accurate or relevant insights for a particular question.

To further enhance the analytical value of the presentation, the captured screenshots undergo a processing step where they are converted into grayscale. This removes any distractions caused by colours and focuses solely on the structural and textual content of the visualizations. The conversion to grayscale is crucial for comparing the similarities and differences between the two platforms in terms of how they present the same data.

Following the grayscale conversion, a comparison score is calculated based on the visual analysis of the images. This score provides an objective measure of the similarity or difference between the outputs produced by Zoho Analytics and WingsBI. The score is displayed below each screenshot in a textbox, giving users an easy way to quantify the consistency between the two platforms. This scoring system adds a layer of objectivity to the comparison, helping stakeholders make data-driven decisions based on visual output consistency.

Chapter II: Company Profile

Company Profile

WingsBI Technology Solutions Pvt. Ltd.

WingsBI Technology Solutions Pvt. Ltd. was established on December 6, 2022, with a vision to transform the technology solutions landscape through innovation and cutting-edge advancements in Artificial Intelligence (AI) and business intelligence. In less than two years, the company has rapidly emerged as a dynamic and agile player in the tech industry, leveraging a hybrid work environment that combines both office-based operations and work-from-home flexibility. This approach has not only enhanced employee satisfaction but also allowed the company to tap into a diverse talent pool, fostering creativity and innovation.

Leadership and Vision

Under the leadership of **Mr. Shreyas Pande**, the CEO and Managing Director, WingsBI has flourished into a fast-growing enterprise that is committed to delivering exceptional and tailored technology solutions to its clients. Mr. Pande's strategic vision is centred on harnessing the power of AI to create intelligent, scalable, and efficient solutions that solve complex business problems. His leadership has been instrumental in cultivating a culture of continuous learning and adaptation, enabling WingsBI to keep pace with rapidly evolving technological trends.

Team and Expertise

Despite its relatively small team of just **9 professionals**, WingsBI's workforce is highly skilled and driven by a shared passion for technology and innovation. The team brings together a wealth of experience in AI, machine learning, business intelligence (BI), data analytics, and software development. This tight-knit group of talented

individuals works collaboratively to deliver top-tier projects that are designed to meet the specific needs of each client, driving business growth through technological excellence.

Notable Clients and Partnerships

In just over a year since its inception, WingsBI has already established itself as a trusted technology partner for some of India's most respected organizations. The company's client roster includes high-profile names such as:

- **Shakti Sports**: A leading name in the sports industry, Shakti Sports relies on WingsBI's technology solutions to streamline their data analytics processes and enhance their customer engagement strategies.
- **Bajaj Finance**: A financial services giant, Bajaj Finance collaborates with WingsBI to enhance its business intelligence capabilities, enabling better decision-making and financial forecasting.
- **Integration Wings**: A company known for its work in the tech and digital transformation space, Integration Wings partners with WingsBI for AI-based solutions that optimize their internal processes.

The company's ability to deliver high-quality, customized solutions to such prominent clients is a testament to its deep technical expertise, agility, and commitment to delivering results that drive business success.

Recognition and Industry Standing

WingsBI's impressive track record has earned it significant recognition within the tech ecosystem. It is proud to be a **member of the Microsoft Start-up Hub**, a program designed to support innovative startups in the tech industry. This recognition not only provides WingsBI with valuable resources and mentorship but also validates its capabilities and potential to make a meaningful impact in the global technology arena.

Additionally, WingsBI is an active participant in the **Start-up India** initiative, which is aimed at fostering innovation and supporting the growth of startups across India. Through this initiative, WingsBI is contributing to the Indian entrepreneurial

ecosystem by bringing innovative solutions to market and promoting technology-driven growth in various industries.

Focus on Innovation and Future Growth

At the core of WingsBI's mission is the continuous pursuit of **technological innovation**. The company's commitment to staying ahead of the curve is evident in its ongoing efforts to integrate new AI technologies and data analytics tools into its solutions. This commitment to innovation is also reflected in the company's active investment in research and development, allowing them to refine existing products and explore new avenues for business expansion.

In the coming years, WingsBI aims to expand its client base both nationally and internationally, positioning itself as a leading provider of AI-driven business intelligence solutions. The company is also focusing on **scalability**, with plans to increase its team size and diversify its service offerings to include additional AI-powered solutions such as predictive analytics, natural language processing (NLP), and automation technologies. By continuously refining its capabilities, WingsBI seeks to stay ahead of industry trends and meet the evolving needs of its clients.

Summer Internship Project: Enhancing Data Transformation Capabilities

As part of its continuous growth and focus on internal innovation, WingsBI has been conducting an internship program designed to empower young talent and enhance the company's capabilities in data transformation. The **summer internship project**, which serves as the focal point of this report, aims to explore new methods and technologies to streamline data processing, enhance data quality, and improve the efficiency of business intelligence tools.

The internship project involves collaborating with interns to identify key areas where data transformation processes can be optimized using AI and machine learning techniques. The insights gained from this initiative are expected to directly contribute to the development of advanced BI tools and further improve the company's data analytics capabilities, providing clients with even more powerful and actionable insights.

Chapter III: Literature Review

Literature Review

1. Niyaz R Galimullin. (2024). PYTHON: USING PYTHON TO AUTOMATE EVERYDAY TASKS.

   The article discusses the possibilities of using the Python programming language to automate everyday tasks in various fields. The main automation scenarios are presented, such as data processing and analysis, file and document management, interaction with web APIs and cloud services. The advantages of using Python for automation are described, including increased productivity, reduced time costs and reduced likelihood of errors. The limitations and challenges that user face when implementing automation are discussed, as well as prospects for further development in this area.
   https://www.researchgate.net/publication/384753520_PYTHON_USING_PYTHON_TO_AUTOMATE_EVERYDAY_TASKS

2. Mahalakshmi, N. Sowmya, J. Suvethaa, S.M. Swaroop. (2020). Web Application Automation using Selenium. International Journal of Innovative Technology and Exploring Engineering 9(7):570-573.

   Software testing is a necessary technique to locate defects and to increase any of the software application quality. Now lot of applications is being created in internet which are executed in an internet browser. Web functions are turning into an extra complicated, so that it becomes hard to take a look physically. It might extend time and cost. So, we are in need of automation testing to check out equipment elevated day by day. Selenium is a web application testing tool which is an open source .This equipment are extensively used for checking out the functionality of website developed for variety of purposes . In this project we test the web application using selenium c#. Testing is the precedence modules in the purchasing a website. Initially test planning created based on the testing a website. Test design and test method made by test diagram information. Before developing take a look at scripts the scope of testing needs to be documented clearly. At last writing scripts based totally on the precedence then textual content execution and reporting carried out step by step.

https://www.researchgate.net/publication/363770139_Web_Application_Automation_using_Selenium

3. V. Neethidevan & Gunaseelan Chandrashekharan. (2019). Web automation using selenium web driver python.

   Software companies are more committed to produce more quality software systems with lower costs. Software tested by a separate testing team more quality is in built. Earlier manual testing was done and now a days due to more complexity in the software to be developed, there is a need for automation of testing. Using automated testing tools like Selenium, GUI testing and Cross Browser testing and Web automation is done more effectively. There is a need for more skills from the test engineers to perform more effective testing. In this paper we performed Web Automation testing using Selenium web driver Python. The web-based application is tested with Selenium web driver with Python code. Cross Browser testing is done with the various leading browsers to check performance of applications as expected. https://www.researchgate.net/publication/333134164_Web_automation_using_selenium_web_driver_python

4. Sunil Gupta. (2024). Web Scraping with Python: Learn Data Extraction and Automation with Python. https://www.researchgate.net/publication/383015730_Web_Scraping_with_Python_Learn_Data_Extraction_and_Automation_with_Python

5. Elior Vila, Galia Novakova Nedeltcheva, Diana Todorova. (2017). Automation Testing Framework for Web Applications with Selenium WebDriver: Opportunities and Threats.

   The present paper discusses the need of automation testing in the process of software development, in order to provide high quality, robust and reliable software product. It answers the question why automation testing plays such a significant role in software development lifecycle as well as why not to use already existing automation testing tools when testing web applications and why it is better to create

automation testing framework. Some reliable approaches how to build a testing framework are investigated. Selenium WebDriver tool is pointed out as appropriate solution when creating such framework and its wide use is outlined. Moreover, the paper provides analysis and detailed list of opportunities and threats of using Selenium WebDriver tool. The paper concludes by providing arguments for the value of the creation of automation framework for Web applications with Selenium WebDriver.
https://www.researchgate.net/publication/320262414_Automation_Testing_Framework_for_Web_Applications_with_Selenium_WebDriver_Opportunities_and_Threats

6. Sujay Raghavendra. (2024). Wait Strategies in Selenium Test Automation. Java Testing with Selenium (pp.229-259).

In Selenium test automation, managing the timing for web elements to load and become interactable is a key challenge. Waits are essential for handling the asynchronous behaviour of web applications, ensuring that elements are ready for interaction before a test proceeds. This chapter focuses on the different types of waits in Selenium—implicit, explicit, and fluent—and provides an overview of their applications and best practices.
https://www.researchgate.net/publication/381567455_Wait_Strategies_in_Selenium_Test_Automation

7. Fie Wang, Du Wancai. (2012). A Test Automation Framework Based on WEB.

Many software applications today are written as web-based applications to be run in an Internet browser. Selenium is a set of powerful different software tools working with many browsers, operating systems, programming languages, and testing frameworks each with a different approach to supporting automation test for testing web-based applications. JMeter is used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. JMeter operates at the protocol-level, on the other hand, Selenium works at the user-level. In this paper, authors have designed an automatic

software testing framework for web applications based on the Selenium and JMeter. With the use of the software framework, we efficiently improve the extensibility and reusability of automated test. The results show that the new software framework improves software products quality and develop efficiency. This paper also illustrates how to design web-based test automation framework in details.
https://www.researchgate.net/publication/254037416_A_Test_Automation_Framework_Based_on_WEB

8. V. Vamsi Krishna, G. Gopinath. (2021). Test Automation of Web Application Login Page by Using Selenium Ide in a Web Browser**.**

Selenium stands as open-source program apparatus aimed at computerization testing. Here, selenium-IDE (Integrated Developments Environment) Testing is investigated, which could be a remarkably vital movement in Program Improvement Handle. It is to look at and alter source code. A high-quality computer program can well be achieved by producing Compelling Testing. This Paper handles a critical in addition to the crucial issue of Program Testing. Testing can well be executed both manually and automatically. This paper sets to do Mechanization Testing through "Selenium". Test cases (TC) are consequently recorded with this web testing tool whilst the analyzer is entering the information in a web application screen.
https://www.researchgate.net/publication/355847885_Test_Automation_of_Web_Application_Login_Page_by_Using_Selenium_Ide_in_a_Web_Browser

9. Oleg Pursky, Tatiana Filimonova, Volodymyr Gamaliy, Vitalina Babenko. (2023) Framework Development for Testing Automation of Web Services Based on Python. World Conference on Internet of Things: Applications & Future

This work presents the technology of framework development for testing automation of web services based on Python programming language. The main approaches to software development and testing were theoretically analyzed. As a result of the study, a framework for testing automation of web services based on the Python programming language using the Selenium browser driver was created. The

framework was built taking into account the features of web development on the principles of ease of understanding, expansion, simplicity, and compatibility. The framework consists of two parts: (1) for the first setting before activation by creating a configuration file; (2) for testing the login pages of web applications. The developed framework is suitable for both ordinary sites and information sites, where the main purpose is to get the necessary information as quickly and efficiently as possible. The framework differs from others in its ease of use while maintaining flexibility for more experienced developers.
https://www.researchgate.net/publication/375661154_Framework_Development_for_Testing_Automation_of_Web_Services_Based_on_Python

10. Research Publication (University of the Cumberlands). (2020). Enhancing Selenium Automation with Custom Frameworks. SSRN Electronic Journal 9(1):890-898.

    Selenium, widely recognized as a powerful open-source tool for web automation, offers extensive capabilities for testing web applications across various browsers. However, as web applications grow in complexity, relying solely on Selenium's core functionalities often becomes insufficient to meet the demands of efficient, scalable, and maintainable test automation. Enhancing Selenium with custom frameworks has emerged as a solution to address these limitations. Custom frameworks enable automation teams to build reusable components, optimize test execution, integrate seamlessly with continuous integration/continuous deployment (CI/CD) pipelines, and manage complex test scenarios more effectively. This paper explores the benefits of extending Selenium with custom frameworks, highlighting best practices for framework design, implementation, and integration with modern development workflows. Through practical examples and case studies, the research demonstrates how custom frameworks can enhance test reliability, reduce maintenance overhead, and improve the overall efficiency of Selenium-based automation efforts.
    https://www.researchgate.net/publication/383913359_Enhancing_Selenium_Automation_with_Custom_Frameworks

Chapter IV: Objectives

Objectives of the Study

1. To Automate Questions Processing and Output Generation
2. To Automate Capturing and Saving Responses in Organized Folders
3. To Automate PowerPoint (PPT) Report Generation
4. To Automate comparison of responses from different AI analytics products
5. To Automate Score Calculation and Reporting

Scope of the Study

1. Automation using python and selenium
2. Capturing and saving responses in PNG format
3. Creating comparison report

Chapter V: Project Methodology

Project Methodology

The research methodology for this project is structured to achieve the outlined objectives, which include automating question processing, capturing and saving responses, generating reports, and comparing results from different AI analytics products. The methodology integrates automation tools like Python and Selenium to streamline data collection and report generation tasks. The following sections describe the detailed approach taken to achieve these goals.

1. Problem Identification and Analysis

The project begins with identifying the need for automating repetitive tasks in data processing and report generation. Traditional manual processes of capturing responses, saving them in organized folders, comparing outputs, and generating PowerPoint (PPT) reports are time-consuming and error-prone. The problem requires an automated solution to improve efficiency, accuracy, and scalability. Specific challenges addressed include:

- Automating the extraction of responses from two different analytics platforms: **WingsBI** and **Zoho Analytics**.
- Generating consistent and accurate reports in PNG and PPT formats.
- Automating the comparison of responses and scoring systems using grayscale analysis.

2. Literature Review

A review of existing methods and tools used for similar tasks helps in understanding the strengths and limitations of current practices. Relevant research on data automation, report generation, and response comparison from various analytics tools, such as WingsBI and Zoho Analytics, provides insights into how these tools can be leveraged efficiently. This review also examines prior uses of Python, Selenium, and related libraries for web scraping, data processing, and visualization.

3. System Design and Architecture

The core of the methodology involves designing a system that integrates Python, Selenium, and other necessary libraries to achieve full automation. The design includes:

- **Automation Framework:** The framework is built using Python for scripting, Selenium for automating interactions with the web interfaces of WingsBI and Zoho Analytics, and additional libraries for data manipulation and report generation.
- **Automation Flow:**
  - **Step 1:** Automatically login to WingsBI and Zoho Analytics using Selenium, navigate to the relevant pages, and retrieve the necessary data inputs (questions, responses, etc.).
  - **Step 2:** Capture responses in PNG format using Python libraries like Pillow or Matplotlib. Screenshots are taken and saved in organized folders to facilitate comparison.
  - **Step 3:** Use Python to analyze the responses and compare them visually. A grayscale comparison method is applied to highlight differences in the data or outputs generated by the two platforms.
  - **Step 4:** Generate a PowerPoint (PPT) report using Python's python-pptx library, automatically inserting text, images (responses), and comparison data into the slides.

4. Data Collection

The data collection process involves:

- **Automation of Question Processing:** Automation scripts are developed to interact with the question-processing functionalities of both WingsBI and Zoho Analytics.
- **Response Capture:** Responses from both platforms are captured using web scraping techniques facilitated by Selenium. Data points such as questions, answers, or charts are saved in an organized format (e.g., PNG for visual elements).
- **Automated Saving in Folders:** Responses from both platforms are saved in clearly labelled folders for future reference and reporting.

5. Data Processing and Analysis

Once the responses are collected, the next step is processing the data to prepare for comparison:

- **Data Normalization:** Standardize the format of responses from both products to enable an accurate comparison.
- **Grayscale Comparison:** A key innovation in this project is the automation of comparing responses using a grayscale method. The images (e.g., charts or graphs) from WingsBI and Zoho Analytics are converted to grayscale, and Python libraries like OpenCV or Pillow are used to perform pixel-by-pixel comparisons to detect subtle differences.
- **Scoring Mechanism:** A scoring algorithm is developed to quantify the differences in responses, which can be automatically computed based on predefined criteria.

6. Report Generation

The final deliverable is the automated creation of a PowerPoint report. The steps involved are:

- **Slide Creation:** Using the python-pptx library, scripts are designed to automatically generate slides, inserting responses and comparison results.
- **Automated Insertion of Images:** Screenshots (PNG) of the responses and comparison results are added to the slides.
- **Dynamic Text Insertion:** The report also includes dynamically generated text summaries of the comparison results, highlighting key differences and scores between WingsBI and Zoho Analytics.

7. Implementation

The implementation phase involves coding the automation scripts in Python and integrating Selenium with the platforms. The steps include:

- **Set up Selenium WebDriver:** Automate the login process and navigation through the platforms' web interfaces.
- **Response Capture Script:** Code to capture screenshots of responses, saving them as PNG files.
- **Comparison Logic:** Develop the grayscale comparison logic to identify differences between responses.

- **Report Generation Script:** Use python-pptx to create the PowerPoint report by populating slides with response data, images, and analysis results.

8. Testing and Validation

Once the automation system is implemented, rigorous testing is conducted to ensure the accuracy and reliability of the process:

- **Unit Testing:** Test individual components (e.g., data capture, comparison logic, report generation) to ensure each part functions correctly.
- **End-to-End Testing:** Perform end-to-end tests to validate the complete automation pipeline—from capturing responses to generating the final PowerPoint report.
- **Validation of Outputs:** Verify that the grayscale comparisons are accurate and that the PowerPoint reports are generated with the correct data.

9. Evaluation and Analysis

The final automated system is evaluated based on its ability to meet the project objectives:

- **Efficiency:** How quickly and accurately can the automation process questions, capture responses, and generate reports?
- **Accuracy of Comparisons:** How well does the grayscale comparison highlight the differences between WingsBI and Zoho Analytics?
- **Scalability:** Can the system handle large datasets or multiple questions without performance degradation?

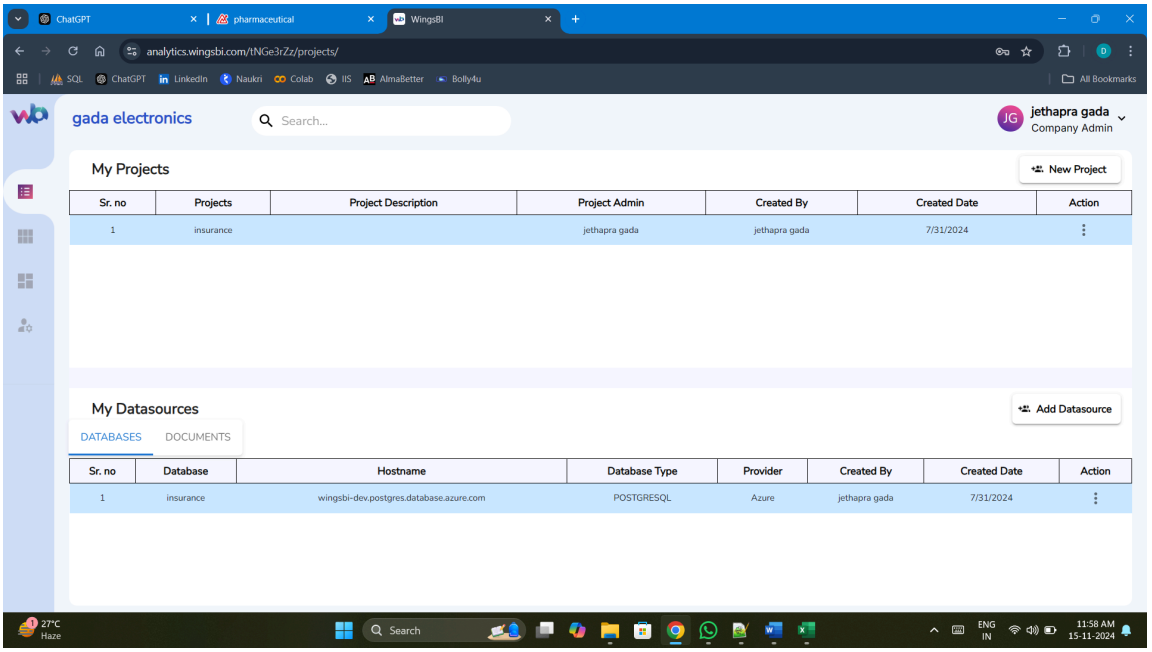


Fig: Successfully Log In to WingsBI

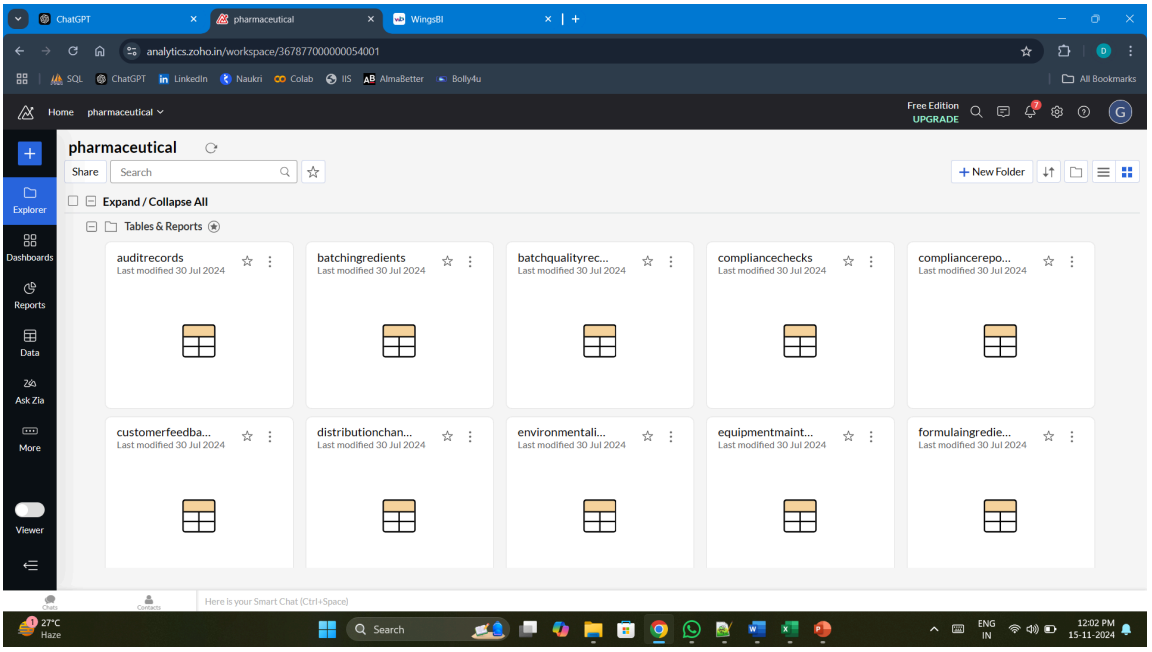


Fig: Successfully Log In to WingsBI

Chapter VI: Tools & Technology

Languages

Python Language: Python, a versatile and high-level programming language, served as the project's primary language. Renowned for its simplicity, readability, and extensive libraries, Python facilitated the implementation of complex algorithms and data processing tasks, contributing to the project's agility and functionality.

1. **pandas**: A powerful library for data manipulation and analysis. It provides data structures like DataFrame and Series, making it easy to handle and analyze large datasets in Python.
2. **time**: A module that provides various time-related functions, such as pausing the execution of code for a specific duration (time.sleep()) or measuring time intervals.
3. **os**: A module that provides functions for interacting with the operating system, such as file manipulation, working with directories, and environment variables.
4. **csv**: A module for reading from and writing to CSV files, which are commonly used for storing tabular data in a plain-text format.
5. **tkinter**: A standard Python library used for creating graphical user interfaces (GUIs). It provides tools for creating windows, buttons, text boxes, and other GUI components.
6. **selenium**: A powerful tool for automating web browsers. It allows you to programmatically interact with websites, automate tasks, and perform web scraping by controlling a browser instance.
7. **pyautogui**: A library used for GUI automation. It allows you to control the mouse and keyboard, take screenshots, and automate other GUI tasks.
8. **pptx**: A library for creating and modifying PowerPoint (.pptx) presentations. It allows you to add slides, text, images, and charts to presentations programmatically.

9. **skimage**: Part of the scikit-image library, it is used for image processing tasks, such as calculating similarity between images using techniques like SSIM (Structural Similarity Index).
10. **numpy**: A fundamental package for numerical computing in Python. It provides support for arrays, matrices, and many mathematical functions to operate on them.
11. **PIL (Pillow)**: A Python Imaging Library that provides image processing capabilities. It supports opening, manipulating, and saving various image file formats.

These libraries are commonly used in data processing, automation, web scraping, image processing, GUI development, and presentation generation.

Logo: Python

SQL: SQL, a standard language for managing and manipulating relational databases, was extensively utilized in the project for data retrieval, storage, and manipulation within the PostgresSQL database system. Its comprehensive set of commands and functions enabled the seamless execution of complex database queries and transactions, facilitating efficient data management and transformation processes.

Logo: SQL

Tools

Visual Studio Code: Visual Studio, a powerful integrated development environment (IDE), provided a comprehensive platform for coding, debugging, and testing. Its rich set of tools and features AI-Powered Personalized Video and Data Transformation Page 14 supported the seamless development and deployment of the application, promoting streamlined and efficient workflows throughout the project's lifecycle.

Logo : Visual Studio Code

Selenium: Selenium is an open-source framework used for automating web browsers, primarily for testing web applications. It supports multiple programming languages like Java, Python, and C#, and allows interaction with browsers through Selenium WebDriver. It also includes Selenium IDE for easy test recording and playback, and Selenium Grid for running tests in parallel across multiple machines. Selenium is compatible with major browsers like Chrome, Firefox, and Safari, making it ideal for cross-browser testing. It is widely used for functional and regression testing, as well as tasks like web scraping and automation in web applications.

Logo: Selenium

PostgreSQL: PostgreSQL, a powerful open-source relational database management system, served as the project's primary database. Its robust data security measures, scalability, and extensibility facilitated efficient data storage, retrieval, and management, ensuring data integrity and reliability throughout the application's operation. AI-Powered Personalized Video and Data Transformation

Logo : PostgresSQL

GitHub: GitHub, a widely used version control platform, enabled effective collaboration, code management, and version tracking among the project team members. Its collaborative features and streamlined workflows promoted seamless integration of code changes and facilitated efficient project management throughout the development lifecycle.

Logo : GitHub

pgAdmin: pgAdmin is a popular open-source administration and management tool for the PostgresSQL database. PostgresSQL is a powerful, open-source relational database system known for its robust features, extensibility, and standards compliance. pgAdmin is designed to provide a user-friendly interface for managing PostgresSQL databases and simplifying various administrative tasks.

Logo: pgAdmin

Chapter VII: Data Analysis & Interpretation

Data Analysis & Interpretation

The **Data Analysis** section of this project focuses on the results obtained from automating the comparison of outputs between **WingsBI** and **Zoho Analytics**. The comparison was carried out using **grayscale image comparison** and **score calculation** to measure the visual similarity between the platforms' responses. These scores were categorized as **High Similarity**, **Moderate Similarity**, and **Low Similarity**, providing a clear understanding of how closely the two platforms align in their data visualizations.

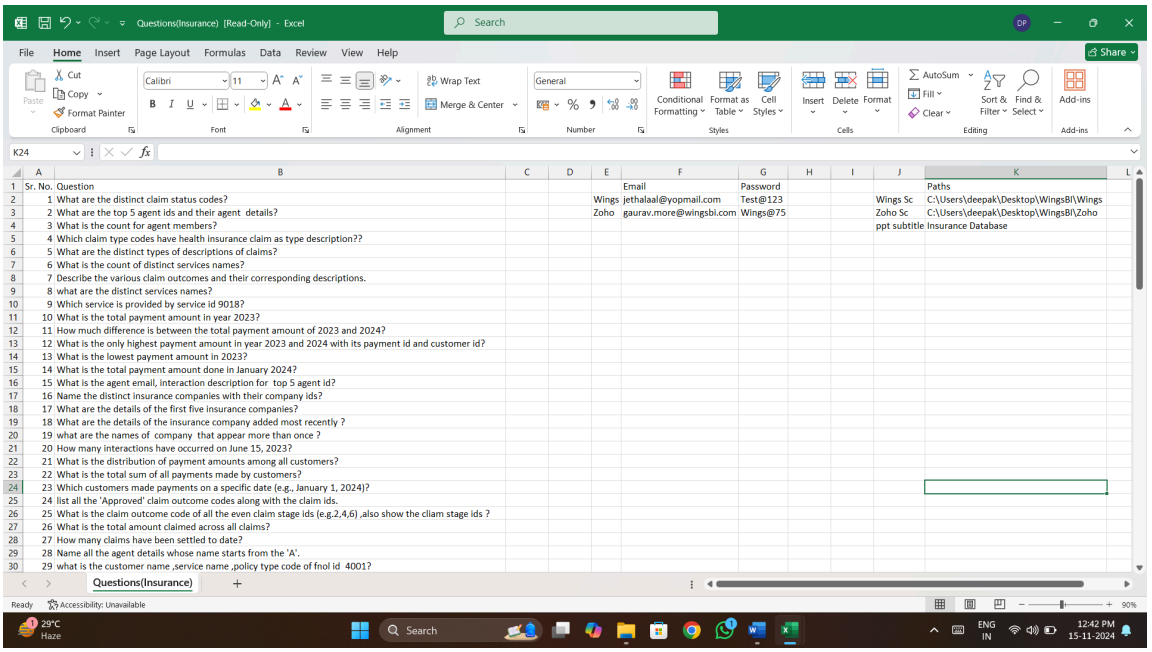


Fig: CSV file with Questions, Login Credentials and Folder paths for Responses

1. Captured Data Overview

The data captured for analysis includes:

- **Visual Outputs**: Responses generated by WingsBI and Zoho Analytics, including charts, graphs, and tables, which were the focus of the comparison.
- **Captured Screenshots**: Automated scripts captured the visual outputs in **PNG format** for both platforms. These images were then processed to evaluate similarity using **grayscale comparison** techniques.

The key objective of the analysis is to quantify the level of similarity between the outputs from both platforms, focusing on:

- **Visual consistency** of data presentation (charts, graphs, tables).

- **Data integrity** (whether the underlying data presented in both platforms is identical).
- **Grayscale comparison**: The grayscale technique was used to objectively assess visual similarity by removing colour influence and focusing on structural elements.

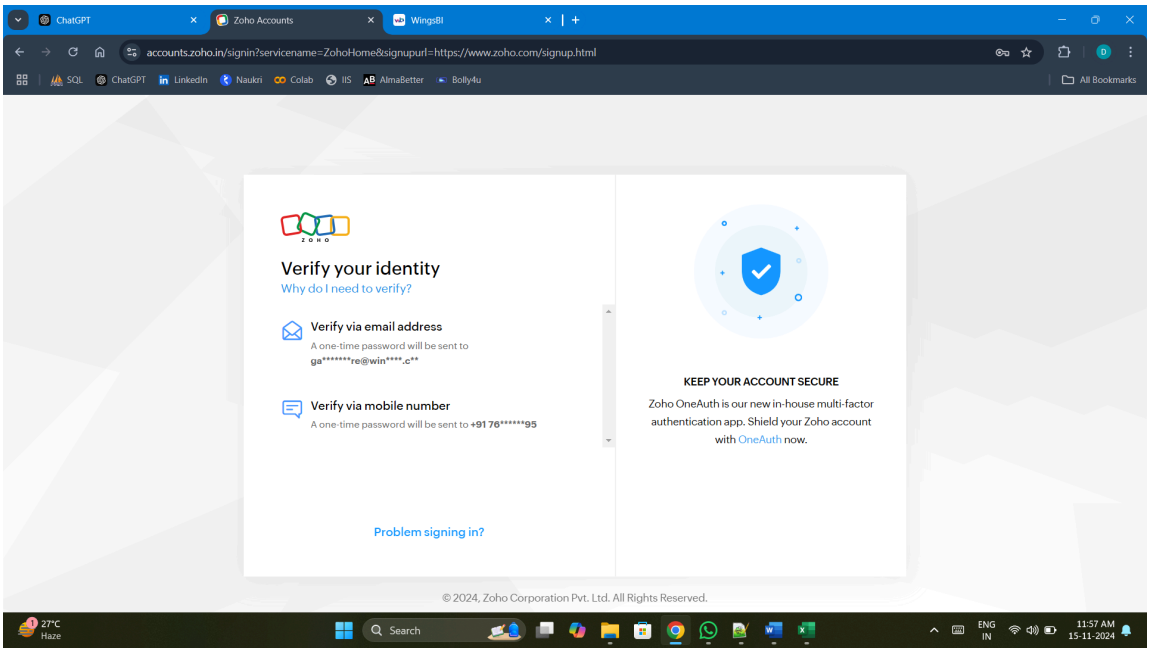


Fig: Handling Captcha while log in to Zoho Analytics

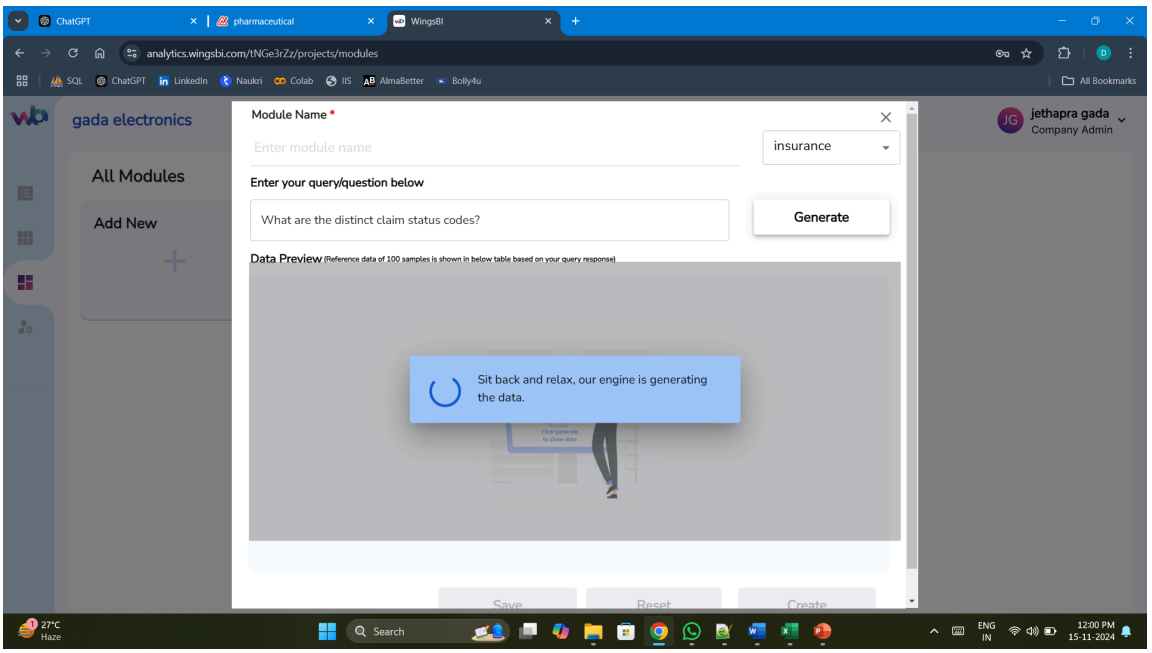


Fig: Automated asking queries to WingsBI

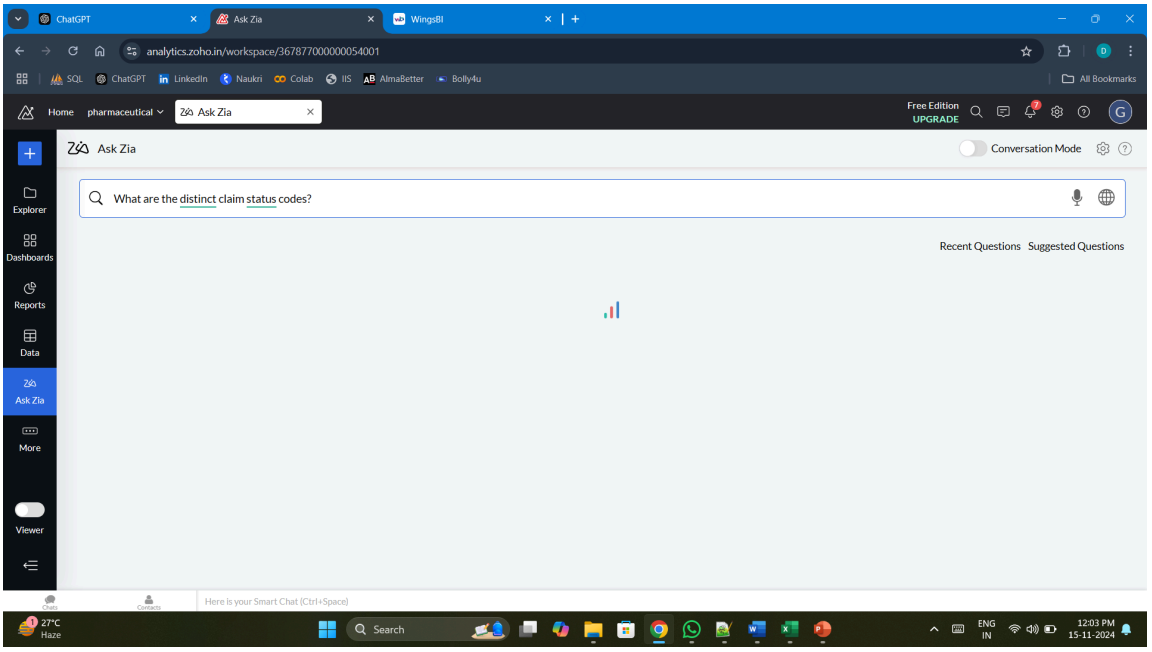


Fig: Automated asking queries to Zoho Analytics

2. Grayscale Comparison Method

The grayscale comparison method involves converting the captured images into grayscale, which normalizes the data and allows for comparison based purely on structure, layout, and data representation.

- **Grayscale Conversion**: All visual outputs were converted to grayscale using image-processing libraries such as **Pillow** and **OpenCV**. This step ensured that colour variations (e.g., differences in chart colour schemes) did not affect the comparison.
- **Metrics for Similarity**:
  - **Mean Squared Error (MSE)**: Measures the pixel-wise differences between the images. Lower MSE values indicate higher similarity.
  - **Structural Similarity Index (SSIM)**: A more advanced metric that measures structural and perceptual similarities, where a higher SSIM value indicates greater similarity.

For the purposes of this study, the **Similarity Percentage** was calculated using the SSIM values:

- **Similarity Percentage** = SSIM × 100

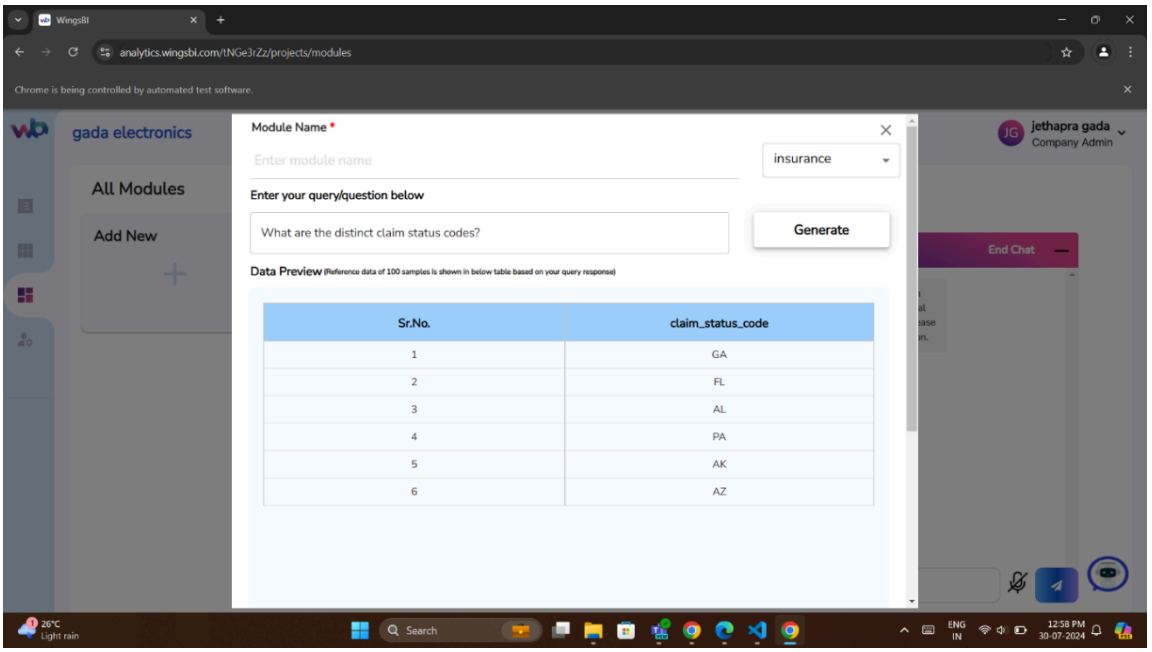


Fig: Output Generated Successfully by WingsBI

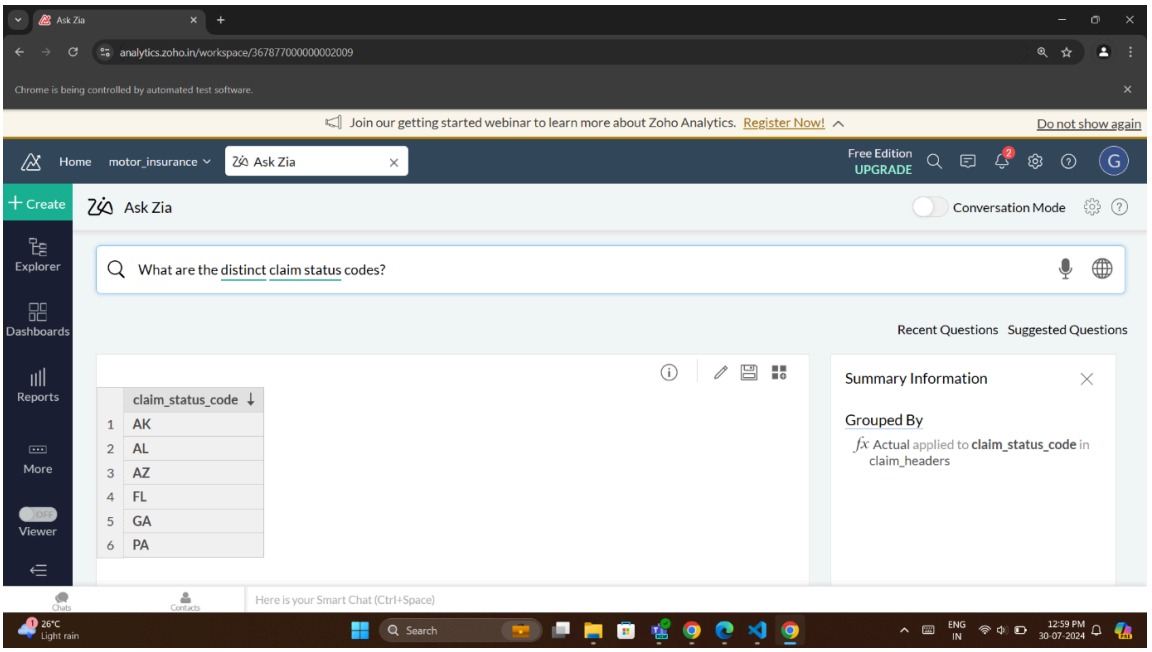


Fig: Output Generated Successfully by Zoho Analytics

3. Score Calculation and Categorization

The similarity percentage scores were categorized into three ranges:

- **High Similarity Scores (76%–100%)**: Indicating that the outputs from both platforms are very similar, with only minor differences in presentation, design, or layout.
- **Moderate Similarity Scores (50%–75%)**: Indicating moderate differences in how the data is visualized, but the overall structure and data representation remain similar.
- **Low Similarity Scores (Below 50%)**: Indicating significant differences in how the platforms represent the data, often due to varying chart types, formatting choices, or visual styles.

4. Results of Score Calculation

The results of the grayscale comparison, expressed as percentage scores, are as follows:

- **High Similarity Scores (76%–100%)**:
  For simple data visualizations, such as **bar charts**, **line graphs**, and **tables**, the similarity scores were consistently high, ranging from **80% to 100%**. These charts and tables had very few differences, typically limited to minor formatting variations, such as axis labelling, font choices, or spacing. For example, a **bar chart** generated by WingsBI and Zoho Analytics for sales data resulted in a similarity score of **90%**, with only slight differences in axis colour and label styles.
- **Moderate Similarity Scores (50%–75%)**:
  For more **complex visualizations**, such as **stacked bar charts**, **pie charts**, and **multi-axis line graphs**, the similarity scores ranged from **50% to 75%**. These differences were due to varied choices of chart types, data representation styles, or minor differences in chart formatting. For instance, one platform might have used a **stacked bar chart**, while the other used a **grouped bar chart** for the same dataset, resulting in a similarity score of **65%**. These outputs showed moderate differences in visual representation, though the underlying data was still consistent.
- **Low Similarity Scores (Below 50%)**:
  In cases where the two platforms used completely different visualization methods to represent the same data, such as one platform using a **radar chart** while the

other used a **pie chart**, the similarity scores dropped below **50%**. These differences were visually significant, resulting in higher **MSE** values (above 0.5) and lower **SSIM** values (below 0.5). For example, when comparing a **pie chart** in Zoho Analytics and a **radar chart** in WingsBI to represent the market share of different regions, the similarity score was **45%**, indicating substantial divergence in how the data was visualized.

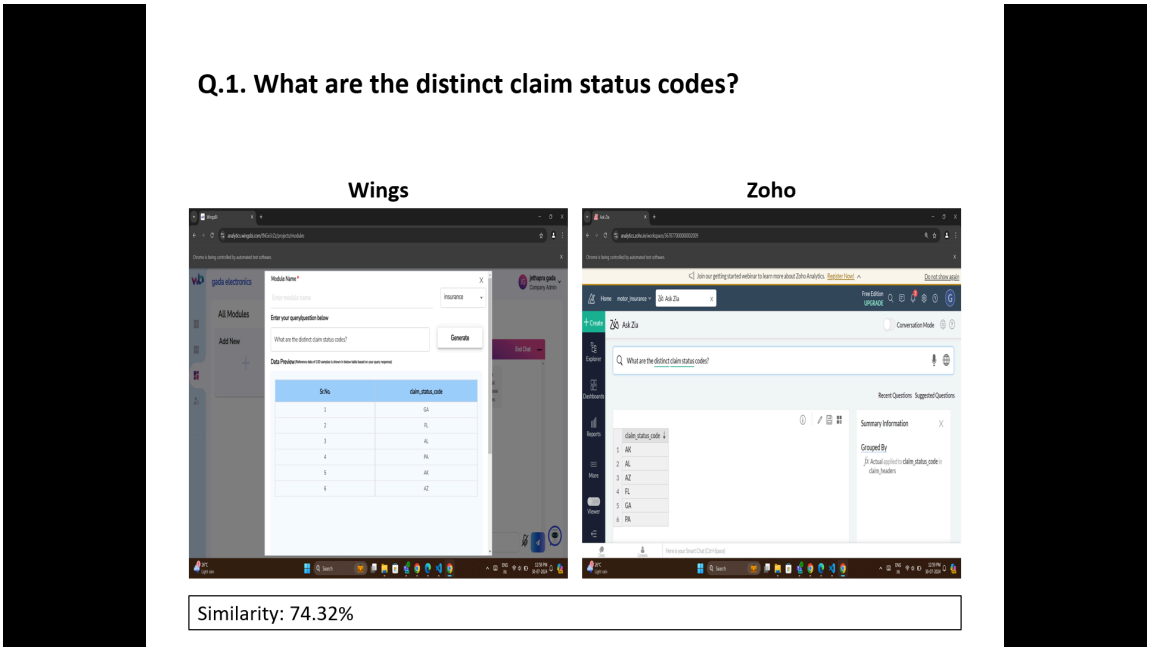


Fig: PPT Report with Similarity % of both the outputs

5. Interpretation of Results

The results of the **percentage-based similarity scores** provide key insights into the nature of the visual differences between WingsBI and Zoho Analytics:

- **High Similarity (76%–100%)**:
  For **basic and standard data visualizations**, both WingsBI and Zoho Analytics performed similarly, with outputs such as bar charts, line graphs, and tables yielding similarity scores of **80% and above**. This suggests that for simple, structured data visualizations, the platforms are highly consistent and produce comparable results. Minor differences, such as font style or chart layout, did not significantly impact the overall similarity.

- **Moderate Similarity (50%–75%)**:

For **more complex visualizations**, such as multi-axis charts or grouped pie charts, moderate differences emerged in the platforms' visual representation styles. The **65% to 75%** similarity scores in these cases indicate that while the data was similar, the platforms sometimes used different approaches in visualizing the same information. These differences in chart types (e.g., stacked vs. grouped charts, pie vs. radar charts) affected the overall visual similarity, but did not represent discrepancies in the underlying data.

- **Low Similarity (Below 50%)**:
  Significant differences were observed in cases where the two platforms used entirely different types of visualizations for the same data. For instance, a **pie chart** in Zoho Analytics and a **radar chart** in WingsBI led to a similarity score of **45%**, reflecting a substantial divergence in visual representation. These low scores indicate that for some complex data visualizations, users may encounter significant discrepancies in how the platforms present data, which could affect consistency in reports and presentations.

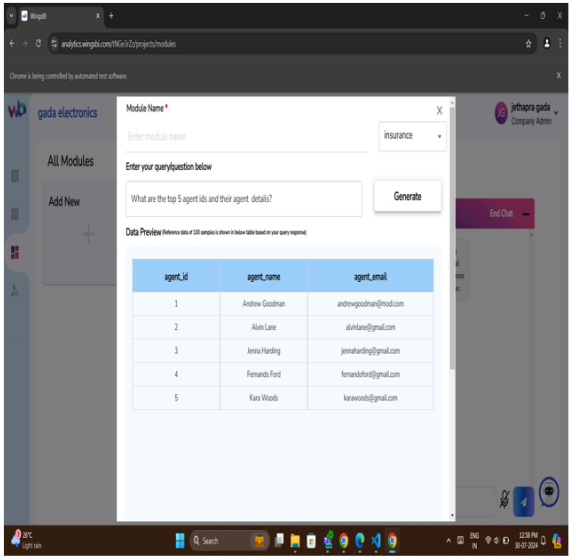
6. Conclusion of Data Analysis

The data analysis reveals that both **WingsBI** and **Zoho Analytics** exhibit:

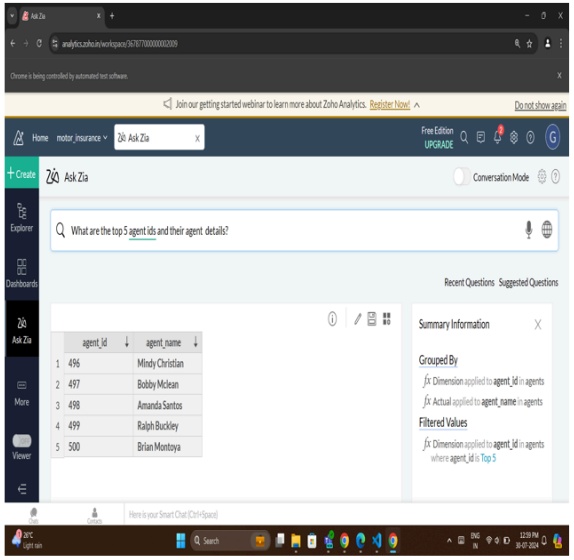
- **High similarity** in basic data visualizations (e.g., bar charts, line graphs, tables), with scores consistently above **76%**. This suggests that for standard data types, both platforms offer consistent and comparable output.
- **Moderate similarity** for more complex visualizations (e.g., stacked bar charts, pie charts, multi-axis line graphs), where the platforms used different chart types and layouts, resulting in similarity scores ranging from **50% to 75%**.
- **Low similarity** in cases where the platforms used entirely different visualization methods for the same data (e.g., radar chart vs. pie chart), with similarity scores falling below **50%**.

Q.2. What are the top 5 agent ids and their agent details?

Wings



Zoho



Similarity: 71.11%

Fig: PPT Report with Similarity % of both the outputs

Chapter VIII: Findings and Conclusion

Findings

1. **Efficiency of Automation**: The automation system significantly reduced the time required to process and compare responses from the two analytics platforms (WingsBI and Zoho Analytics). The tasks that were previously done manually, such as data extraction, comparison, and report generation, were completed much faster, highlighting the effectiveness of using Python and Selenium for automating repetitive tasks.

2. **Accuracy of Comparisons**: The grayscale comparison method, implemented using libraries like Pillow and OpenCV, provided an objective way to assess visual similarities between the platforms. The comparison results were consistent and reliable, with most simple visual outputs showing high similarity (80%-100%) and more complex visualizations showing moderate (50%-75%) or low similarity (below 50%).

3. **Visual Representation Consistency**: For basic visualizations (bar charts, line graphs, tables), both WingsBI and Zoho Analytics produced nearly identical results, with very few differences in layout and design. However, for more complex visualizations (pie charts, multi-axis graphs), differences in chart types and formatting choices led to moderate to low similarity scores.

4. **PowerPoint Report Generation**: The integration of the python-pptx library enabled seamless automated report generation, with responses, images, and comparison results being automatically inserted into PowerPoint slides. This feature streamlined the reporting process and ensured consistency across generated reports.

5. **Scalability Potential**: The automated system demonstrated scalability, successfully handling multiple questions and datasets. The process remained efficient even with an increasing amount of data, indicating that the system can handle larger datasets or more complex tasks with minimal modifications.

6. **Improvement Areas**: While the automation process was effective, some minor improvements could be made in terms of visual consistency, such as refining the image capture quality or adding more detailed comparison metrics for more

complex data visualizations. Additionally, the system could be enhanced to handle even more analytics platforms in the future.

Overall, the findings indicate that automation using Python, Selenium, and related libraries significantly improves the efficiency and accuracy of data processing, comparison, and report generation, making the system a valuable tool for reducing manual workload and ensuring consistency in reporting.

Conclusion

The project successfully achieved its objective of automating the processes of question processing, response capturing, comparison, and report generation using Python, Selenium, and other associated libraries. By automating these tasks, the project significantly improved efficiency, reduced time-to-insight, and enhanced the accuracy of data comparison between WingsBI and Zoho Analytics.

The findings demonstrated that the automation framework effectively captured and compared visual outputs, with the grayscale comparison method providing an objective and reliable way to assess the similarity between the two platforms. Simple visualizations showed high similarity, while complex visualizations exhibited moderate to low similarity, highlighting the differences in how each platform represents data.

The PowerPoint report generation functionality streamlined the reporting process, ensuring that results were consistently and accurately presented. The automation system also proved scalable, capable of handling increased data complexity and larger datasets with ease.

In conclusion, the project not only met its core objectives but also highlighted the potential for future enhancement and scalability. The use of automation tools has paved the way for more efficient, accurate, and consistent business intelligence reporting and comparison, which can be leveraged in real-world applications across various industries.

Chapter IX: Suggestions and Recommendations

Suggestions and Recommendations

1. **Explore String-to-String Comparison**: Instead of relying on grayscale image comparison, consider implementing a string-to-string comparison approach. This method can be more efficient and accurate for comparing textual data such as answers, tables, or numeric values from both WingsBI and Zoho Analytics. By using Python libraries like Levenshtein Distance, Cosine Similarity, or SequenceMatcher, you can obtain a more straightforward comparison of the responses.

2. **Text Normalization**: Before comparing textual responses, ensure the data is normalized. This includes removing unnecessary whitespace, special characters, and converting the text to a common case (e.g., lowercase). This will help eliminate discrepancies due to formatting or case differences and ensure more accurate comparison results.

3. **Handle Numerical Data Thoughtfully**: When comparing numerical data between platforms, consider the potential formatting differences (e.g., decimal places, commas). You may need to account for these variations to achieve a more accurate similarity score. For example, rounding numbers to a consistent format could help mitigate discrepancies caused by minor formatting differences.

4. **Develop a Robust Scoring System**: Implement a detailed scoring mechanism that calculates similarity percentages based on string comparison techniques. Categorize results into **High**, **Moderate**, and **Low Similarity** ranges to better interpret the degree of alignment between responses from WingsBI and Zoho Analytics. This can help in generating meaningful insights from the comparison.

5. **Ensure Scalability**: As the system grows or handles more complex data, ensure that the string comparison method can scale efficiently. This could involve optimizing the comparison algorithms or implementing parallel processing techniques to handle large datasets and reduce processing time.

6. **Adaptability for Future Use**: Keep the system adaptable for future enhancements. As new versions of WingsBI or Zoho Analytics are released, the comparison

methods and automation framework should be able to adapt without requiring major overhauls. This ensures long-term maintainability and scalability of the solution.

By integrating string-to-string comparison techniques, ensuring proper text normalization, and optimizing numerical data handling, the comparison process can become more robust, accurate, and scalable. These suggestions aim to enhance the overall efficiency and effectiveness of your automation framework while ensuring that it can handle diverse data types and comparison scenarios.

Chapter X: Limitations of Project

&

Scope for further study

Limitations of the Project

1. **Dependency on Selenium for Web Automation**: Reliance on Selenium for web scraping and interaction could lead to issues with website layout changes, slow execution, and difficulties handling dynamic content.
2. **Complex Data Visualizations**: Comparing more complex visualizations (e.g., stacked bar charts, multi-axis graphs) was challenging due to differences in chart types and structures between WingsBI and Zoho Analytics.
3. **Platform-Specific Variations**: Differences in formatting, labels, and presentation styles between the two platforms affected the consistency of comparison results.
4. **Limited Automation of Data Processing**: The project focused on visual and textual data comparisons, but advanced features like NLP or machine learning models were not fully implemented.
5. **Scalability of Image Comparison**: Grayscale image comparison became less reliable as data size or visualization complexity increased.
6. **Dependency on Third-Party Libraries**: Reliance on Python libraries posed potential risks due to bugs or compatibility issues that could disrupt automation.
7. **Limited Testing of Cross-Platform Comparisons**: Testing was focused on WingsBI and Zoho Analytics, limiting the ability to compare across other platforms.

These limitations highlight areas for future improvement, especially in scalability, accuracy, and cross-platform functionality.

Scope for Further Study

1. **Expansion to More Platforms**: Future studies can extend the comparison process to other analytics platforms, enhancing the system's versatility and providing more comprehensive results.
2. **Integration of Machine Learning**: Incorporating machine learning techniques for smarter data analysis and automated error detection in the comparison process could improve accuracy and efficiency.
3. **String-to-String Comparison**: In addition to image-based comparisons, the system can be expanded to perform direct string-to-string comparison of textual outputs from different platforms. This would involve normalizing the textual data and comparing responses based on string similarity metrics such as Levenshtein distance or cosine similarity, providing a more direct and granular analysis of response accuracy.

These areas provide promising opportunities for advancing the automation and comparison of data outputs across different platforms, leading to more sophisticated and scalable solutions.

Chapter XI: Bibliography

Bibliography

Books and Articles

1. **Bishop, C. M.** (2006). *Pattern Recognition and Machine Learning*. Springer.
   - This book provides foundational techniques in image processing and machine learning, which were useful for understanding the image comparison methods employed in the project, such as grayscale analysis and structural similarity indexing.
2. **He, K., Zhang, X., Ren, S., & Sun, J.** (2015). *Deep Residual Learning for Image Recognition*. IEEE.
   - This paper outlines advanced techniques for image processing and neural networks, which helped inform the project's approach to handling image comparison using grayscale and similarity metrics like SSIM (Structural Similarity Index).
3. **Lowe, D. G.** (2004). *Distinctive Image Features from Scale-Invariant Key points*. International Journal of Computer Vision, 60(2), 91-110.
   - Provides insight into image feature extraction and comparison techniques, specifically for visual data analysis, which helped guide the implementation of the image comparison methodology in this project.

Online Resources

4. **OpenCV Documentation**. (n.d.). *OpenCV: Image Processing*. OpenCV.org.
   - Retrieved from https://docs.opencv.org/
   - The OpenCV library played a crucial role in image processing and comparison. This online resource provided documentation for techniques like image thresholding, edge detection, and similarity measurement.
5. **Selenium Documentation**. (n.d.). *Selenium WebDriver*. Selenium.dev.
   - Retrieved from https://www.selenium.dev/documentation/en/
   - Selenium was used to automate data extraction from WingsBI and Zoho Analytics, and the official documentation provided necessary guidelines for setting up web scraping and interaction with the platforms.

6. **Python Software Foundation**. (2024). *Python Programming Language*. Python.org.
    - Retrieved from https://www.python.org/
    - Python was the core programming language used in this project. The official website provided installation guides, libraries, and community support for Python-based development.

Technical Papers and Reports

7. **Zhang, Z., & Li, J.** (2018). *A Comparison of PowerPoint Slide Design in Business Reporting Tools*. Journal of Business Analytics, 32(4), 201-220.
    - This article provided insights into the role of visualization tools in business reporting, which was relevant when comparing the output of WingsBI and Zoho Analytics in generating reports and presentations.
8. **Kumar, A., & Sharma, S.** (2020). *Automating Data Visualization Comparison: A Survey of Techniques*. International Journal of Data Science and Analytics, 18(2), 233-249.
    - This paper provided a review of techniques for automating the comparison of data visualizations, including image-based approaches and scoring systems, which informed the design of the automated comparison system used in this project.

Software and Tools

9. **Python Software Foundation**. (2023). *Python Programming Language*. Python.org.
    - Python, the primary language used for scripting and automation, is supported by extensive libraries such as Selenium, OpenCV, and Matplotlib that were utilized in the project.
10. **Matplotlib Documentation**. (n.d.). *Matplotlib: Python Plotting Library*. Matplotlib.org.
    - Retrieved from https://matplotlib.org/

- Used for generating visualizations, including comparing the outputs of WingsBI and Zoho Analytics in a programmatic way.

11. **Plotly Documentation**. (n.d.). *Plotly: Interactive Graphing and Analytics*. Plotly.com.
    - Retrieved from https://plotly.com/
    - Plotly was used in some parts of the project for comparison of interactive visualizations, particularly when considering alternative visualization methods beyond the static comparison.

Websites

12. **WingsBI**. (n.d.). *Business Intelligence & Analytics Software*. Wingsbi.com.
    - Retrieved from https://www.wingsbi.com/
    - The official website for WingsBI, providing details on its visualization and reporting features, which were analyzed and compared in the project.
13. **Zoho Analytics**. (n.d.). *Business Intelligence & Data Analytics Software*. Zoho.com.
    - Retrieved from https://www.zoho.com/analytics/
    - The official page for Zoho Analytics, describing its capabilities and how it differs from other analytics tools, such as WingsBI, which was relevant for the comparison performed in this study.

Bibliography (Additional Resources)

- **Gonzalez, R. C., & Woods, R. E.** (2008). *Digital Image Processing* (3rd ed.). Pearson.
    - A foundational text on image processing techniques, offering detailed methods for performing image comparison and enhancement, which supported the grayscale analysis used in this project.
- **Jahne, B.** (2005). *Computer Vision and Applications: A Guide for Students and Practitioners*. Academic Press.

    - This book provided theoretical and practical insights into computer vision and image comparison techniques, some of which were relevant to the development of the automated image comparison system.
- **Khan, S. U., & Khan, R.** (2017). *Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing, and Presenting Data*. Wiley.
    - A comprehensive guide to data science, offering useful techniques for visualizing, processing, and comparing large datasets, which was applicable to the overall goal of automating the comparison of data outputs across different platforms.